Необходимо найти коэффициент регрессии, это можно сделать при помощи метода наименьших квадратов – один из методов теории ошибок, предназначенный для оценки неизвестных величин по результатам их измерений, содержащим случайные ошибки. Пример кода на *Python* реализующий поиск коэффициентов, при помощи библиотеки *numpy* представлен на рисунке 2.

```
X = data[['Days', 'Low', 'High', 'Open']].values
Y = data['Close'].values

# Вычисление коэффициентов множественной линейной регрессии
A = np.vstack([X.T, np.ones(len(X))]).T
coefficients  = np.linalg.lstsq(A, Y, rcond=None)[0]

print(coefficients)

b1 = coefficients[0]
b2 = coefficients[1]
b3 = coefficients[2]
b4 = coefficients[3]

b0 = coefficients[4]
```

Рис 2. пример вычисления коэффициентов множественной линейной регрессии

Теперь, зная коэффициенты регрессии можно предсказывать значение параметра «*Close*» на определённую дату и с заданными параметрами «*Low*», «*High*», «*Open*», подставив их значения и коэффициенты в формулу 1.

Заключение

Результатом проделанного работы является нахождение коэффициентов регрессии, которые, в свою очередь, позволяют предсказывать параметр «*Close*» на определённую дату.

АЛГОРИТМЫ СРАВНЕНИЯ ТЕКСТОВЫХ ДОКУМЕНТОВ

Кротов А.В. (студент гр. ИТИ-41)

Гомельский государственный технический университет имени П.О.Сухого, Гомель, Республика Беларусь

Научный руководитель – **Курочка К.С.**

(к. т. н., заведующий кафедрой «Информационные технологии» ГГТУ им. П.О. Сухого)

Аннотация: исследуются современные методы сравнения текстов и анализируются их преимущества и недостатки.

Ключевые слова: алгоритмы сравнения текстов, обнаружение заимствований, внутренние заимствования.

Введение

Поиск плагиата среди текстовых документов является сложной, но в то же время востребованной задачей, особенно в академической и студенческой средах. Поиск плагиата – это комплекс средств, позволяющих определить степень схожести двух текстовых документов. Актуальность доклада обусловлена следующими факторами: 1) необходимость предотвращения копирования работ, которые были сделаны другими людьми; 2) активное интегрирование современных информационных технологий в образовательный процесс. Целью данного доклада является анализ существующих алгоритмов сравнения оригинального и проверяемого текстов для определения степени плагиата между ними и их применение для построения пользовательских приложений.

Результаты и обсуждение

Для сравнения текстов существует множество подходов, часть из которых основана на вычислении некоторого числа, описывающего степень схожести сравниваемого и оригинального текстов. Примером такого подхода является алгоритм вычисления косинусной меры подобия [1].

Для расчета косинусной меры подобия два текста разбиваются на отдельные слова, которые обрабатываются стандартными методами – приведение в нижний регистр, удаление

знаков препинания, удаление бессмысленных слов, выделение основы слова в однокоренных словах. Далее находится объединение двух наборов слов – это объединение служит общим для двух документов словарем. Для каждого документа создается вектор, элементы которого равны количеству вхождений соответствующего слова в текущий документ. Мерой плагиата в алгоритме является косинус угла между полученными векторами. Исходя из математических свойств косинуса, делается вывод о степени плагиата – если значение косинуса близко к -1, то документы имеют значительные отличия; если значение косинуса близко к 1, то документы являются схожими. Недостатком алгоритма является то, что он не учитывает длины векторов документов – таким образом, при сравнении нескольких документов различной длины с исходным документом алгоритм может выдать одинаковые результаты.

Примером алгоритма, который учитывает, как общее распределение слов, так и длину векторов, является алгоритм *TS-SS* [2]. *TS (triangle area similarity)* – площадь треугольника, который образуется между двумя векторами документов; чем меньше площадь треугольника, тем более два документа похожи друг на друга. *SS (segment area similarity)* – площадь сегмента, радиус которого зависит от расстояния между векторами; чем меньше площадь сегмента, тем более два документа похожи друг на друга. Результат перемножения полученных метрик для схожих документов будет стремиться к 0, а для различных документов – к бесконечности.

Таким образом, алгоритмы применяются для сравнения двух документов и дают предварительную оценку их схожести в зависимости от распределения использованных в них слов. В пользовательском приложении алгоритмы могут быть применены для сравнения заданных двух документов, которые пользователь может выбрать в зависимости от различных параметров.

Другим подходом для сравнения двух текстов является алгоритм *Winnowing* [3]. Метод является усовершенствованием алгоритма *Fingerprint*: оба алгоритма имеют одинаковую последовательность шагов, однако отличаются методом формирования финального вектора, описывающего документ. Оба алгоритма работают с последовательностями символов или слов, которые называются *n*-граммами. Для составления *n*-грамм изначальный текст обрабатывается стандартными методами, после чего он разбивается на группы по *n* символов, либо *n* слов. Для каждой *n*-граммы вычисляется значение хэш-функции, которая принимает на вход *n*-грамму и возвращает некоторое число. После хэширования *n*-грамм для двух документов получаются два вектора хэш-кодов.

Алгоритм *Fingerprint* для генерации финального вектора использует некоторое число, которое является входным параметром алгоритма: результирующий вектор формируется только из тех чисел вектора, которые при делении на данное число в остатке дают ноль. Недостаток такого подхода заключается в том, что он не гарантирует обнаружение общих *n*-грамм: *n*-грамма, общая между документами, обнаруживается алгоритмом только в том случае, если ее хэш равен кратен входному параметру. Алгоритм *Winnowing* для генерации финального вектора использует понятие «окна», которое перемещается по вектору хэшей слева направо с единичным шагом: «окно» имеет размер, заданный в входных параметрах; применяя «окно» к вектору хэшей, из каждого полученного поднабора выбирается наименьший хэш, который затем входит в финальный вектор.

Полученные векторы являются краткими описаниями документов и могут быть применены для различных целей:

– расчет метрик для оценки степени заимствования: коэффициент Жаккара, коэффициент Сёренсена, косинусная мера подобия;

– сохранение векторов в базу данных для быстрого поиска схожих документов.

Дальнейшим развитием алгоритма является сохранение информации о расположении в оригинальном тексте обрабатываемых *n*-грамм. При совпадении хэш-кодов в двух векторах данная информация может быть использована для определения участка текста, который был

скопирован в сверяемый документ; найденный участок текста может быть визуально выделен для отображения пользователю, сверяющему документ.

Алгоритм является более затратным по времени и памяти по сравнению с алгоритмами вычисления косинусной меры сходства и *TS-SS*, поскольку он занимается дополнительной обработкой выделенных *n*-грамм; с другой стороны, алгоритм обеспечивает более стабильную и точную оценку плагиата. Также помимо оценки степени плагиата в виде числа, алгоритм *Winnowing* решает ряд задач, решение которых может быть полезно для построения пользовательских приложений – задача быстрого поиска схожих документов среди набора всех доступных документов, задача явного указания места плагиата в тексте.

Заключение

Рассмотрены основные алгоритмы для сравнения текстовых документов. Для более быстрого и поверхностного сравнения документов для них могут быть рассчитаны различные числовые метрики, к которым относятся косинусная мера схожести, результат алгоритма *TS-SS*. Для более детального сравнения документов могут быть использованы *Fingerprint* алгоритмы – наиболее стабильным является алгоритм *Winnowing*. С точки зрения создания пользовательских приложений алгоритмы используются для решения различных прикладных задач – сравнение двух заданных документов, быстрый поиск схожих документов, выделение заимствованного участка текста в сравниваемом документе.

Литература

1. Khuat, Tung & Hung, Nguyen & Thi My Hanh, Le. (2015). A Comparison of Algorithms used to measure the Similarity between two documents. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET). 4. 1117-1121.

2. Heidarian, Arash & Dinneen, Michael. (2016). A Hybrid Geometric Approach for Measuring Similarity Level Among Documents and Document Clustering. 142-151. 10.1109/BigDataService.2016.14.

3. Saul Schleimer, Daniel S. Wilkerson, and Alex Aiken. 2003. Winnowing: local algorithms for document fingerprinting. In Proceedings of the 2003 ACM SIGMOD international conference on Management of data (SIGMOD '03). Association for Computing Machinery, New York, NY, USA, 76–85. https://doi.org/10.1145/872757.872770

ТЕХНОЛОГИИ РАСПОЗНАВАНИЯ ЛИЦ ДЛЯ ИДЕНТИФИКАЦИИ СТУДЕНТОВ: АНАЛИЗ И ПЕРСПЕКТИВЫ

Кулаковский Д.В. (студент гр. ИТИ-41)

Гомельский государственный технический университет имени П.О.Сухого, Гомель, Республика Беларусь

Научный руководитель – **Курочка К.С.**

(к. т. н., заведующий кафедрой «Информационные технологии» ГГТУ им. П.О. Сухого)

Аннотация: исследуются современные методы распознавания лиц и анализируются их преимущества и недостатки, а также перспективы использования для идентификации студентов.

Ключевые слова: искусственный интеллект, распознавание лиц.

Введение

Технологии распознавания лиц (*face recognition technologies, FRT*) – это комплекс программных и аппаратных средств, позволяющих идентифицировать человека по его лицу. Актуальность доклада обусловлена следующими факторами: 1) необходимость обеспечения качества и эффективности образовательного процесса, а также предотвращения мошенничества и нарушений академической честности; 2) возрастание интереса к технологиям распознавания лиц как одному из самых перспективных и инновационных направлений искусственного интеллекта и машинного обучения; 3) существование различных подходов к распознаванию лиц, имеющих свои преимущества и недостатки и требующих сравнительного анализа и оценки. Целью данного доклада является анализ существующих технологий распознавания лиц, которые можно использовать для